INTEGRATING EQUATIONS OF MOTION AS AN OPTIMIZATION PROBLEM

James L. Carr, Ph.D.
Carr Astronautics Corporation
1216 16th St. NW
Washington, D.C. 20036
jimcarr@carrastro.com

ABSTRACT

This paper shows how mathematics that is usually applied to optimization and estimation problems can be applied to find numerical solutions of ordinary differential equations. The method is illustrated for the problem of integrating the equations of motion for a nearly geostationary satellite, but can be more generally applied to other problems. The method is interesting because its numerical accuracy can be set in accordance with the uncertainty in the force model. It is also especially well suited to solving differential equations with boundary conditions or constraints that are more complicated than those of the initial value problem.

INTRODUCTION

In early 1992, the author needed a method of propagating the ephemeris of the METEOSAT geostationary weather satellite for an application called the Aerospatiale METEOSAT Image Processing System (AMIPS)[1]. AMIPS required a computationally efficient method that did not require that evaluations be performed in strict chronological order. The solution adopted at the time was to tabulate an ephemeris using PEPSOC[2] and then to fit the table of position vectors to a finite series of Chebyshev polynomials. The polynomials could be rapidly evaluated using numerically stable recursion relations and the series could be evaluated at arbitrary times, in any order, within a defined span. It occurred to the author at the time that the intermediate step of tabulating the ephemeris could be eliminated and the problem of solving the differential equations of motion could be recast in terms of a constrained optimization problem:

> *Given a time span T and a finite series of Chebyshev polynomials, find the coefficients of the series that best represent the solution to the differential equations of motion subject to a sufficient set of constraints such as fixing the initial values of position and velocity.*

Since this time, the author's company, Carr Astronautics, has fully developed this concept, which has become the backbone of all of our proprietary orbit propagation and determination software, including a product line of fully validated simulation and operational software for the GOES and MTSAT programs. The paper begins by developing the method for the problem of integrating an orbit for a nearly geostationary satellite. The paper closes with a discussion of some of the advantages of the method and how it could be applied to other problems.

NOTATION

Boldface letters are used to represent vectors in this paper. Scalars and matrices are represented by italicized letters.

EQUATION OF MOTION FOR A NEARLY GEOSTATIONARY SATELLITE

The motion of a nearly geostationary satellite is best described in an Earth-Centered Earth-Fixed (ECEF) rotating frame, in which the equation of motion for the satellite position is

$$\ddot{\mathbf{R}} + 2\mathbf{\Omega}\times\dot{\mathbf{R}} + \mathbf{\Omega}\times(\mathbf{\Omega}\times\mathbf{R}) = -\frac{GM}{|\mathbf{R}|^3}\mathbf{R} + \mathbf{a}_{perturb}(\mathbf{R},t) \tag{1}$$

where $\mathbf{R}(t)$ is the coordinate position vector and $\mathbf{\Omega}$ is the Earth rate vector (defining the ECEF z-axis). The second and third terms on the left-hand side represent the Coriolis and centrifugal accelerations that arise when working in a non-inertial frame. The acceleration model appears on the right-hand side of equation (1), with the acceleration due to the point-mass attraction of the Earth (GM is the product of the Newtonian gravitational constant with the mass of the Earth) plus a term that represents all the important perturbations on a nearly geostationary satellite:

- Solar radiation pressure
- Non-spherical mass distribution of the Earth
- Lunar and solar point-mass attractions

Other effects that are generally smaller and are not considered further in this paper are:

- Thrust due to RF and thermal self-emissions
- Earth albedo and self-emission radiation pressure
- Lunar illumination radiation pressure
- Point-mass attraction of the other bodies in the solar system
- Temporal variations of the mass distribution of the Earth
- Coriolis and centrifugal accelerations arising from astronomical precession and nutation
- Variations in the Earth rate
- Those of special and general relativity

In the absence of perturbations, equation (1) admits a solution where $\mathbf{R}$ is constant and perpendicular to $\mathbf{\Omega}$.

SOLAR RADIATION PRESSURE

The acceleration due to the pressure of sunlight is most simply represented by a cannonball model:

$$\mathbf{a}_{solar-pressure} = -\frac{I}{c} K \cdot \left(a/m\right)_{effective} \hat{\mathbf{s}} \tag{2}$$

where I is the solar irradiance, c is the velocity of light, K is a shadow factor (zero in eclipse, one in full sun), $\hat{\mathbf{s}}$ is a direction vector towards the sun, and $\left(a/m\right)_{effective}$ is the effective satellite area-to-mass ratio (the physical projected area-to-mass ratio for a perfectly black object and twice that for a perfectly reflective object). More complicated representations of the solar radiation pressure can also be accommodated with a little extra effort.

NON-SPHERICAL EARTH GRAVITY

A non-central component of the gravitational field arises due to the non-spherical mass distribution of the Earth. It may be described by a potential energy function developed in a spherical harmonic series. Our product line incorporates the GEM-T1 model[3] (the reader is referred to the reference for further information).

LUNI-SOLAR GRAVITATIONAL PERTUBATION

The acceleration towards either the moon or the sun is given by

$$\mathbf{a}_{obj} = -\frac{GM_{obj}}{\left|\mathbf{R} - \mathbf{r}_{obj}\right|^3}\left(\mathbf{R} - \mathbf{r}_{obj}\right) - \frac{GM_{obj}}{\left|\mathbf{r}_{obj}\right|^3}\mathbf{r}_{obj} \tag{3}$$

where $\mathbf{r}_{obj}$ is the position vector of either the moon or the sun relative to the Earth. The first term in (3) is the Newtonian gravitational force exerted by the object, whereas, the second term is the acceleration due to the reflex motion of the Earth subject to the gravitational force of the perturbing object. Our product line uses the NOVAS package developed by the U.S.N.O. to compute the positions of the moon and the sun[4].

REPRESENTATION OF THE SOLUTION

The solution to the vector equation of motion (1) on the interval $[0,T]$ is approximated by a Chebyshev polynomial expansion up to a maximum degree D:

$$\mathbf{R}_{app}(t) = \sum_{d=0}^{D} \mathbf{c}_d T_d\left(\frac{2t - T}{T}\right) \tag{4}$$

where $T_n(x)$ is the Chebyshev polynomial of degree n. Chebyshev polynomials and their derivatives (needed to evaluate the velocity and acceleration) are very efficiently evaluated using a ladder of recursion relations[5]:

$$
\begin{aligned}
&T_0(x) = 1 \qquad T_1(x) = x \\
&U_0(x) = 1 \qquad U_1(x) = 2x \\
&T_0''(x) = T_0'(x) = U_0'(x) = 0 \\
&T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \\
&U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x) \\
&T_n'(x) = nU_{n-1}(x) \\
&U_n'(x) = (n+1)U_{n-1}(x) + xU_{n-1}'(x) \\
&T_n''(x) = nU_{n-1}'(x)
\end{aligned}
\tag{5}
$$

Substituting $\mathbf{R}_{app}(t)$ for $\mathbf{R}(t)$ in equation (1) we have

$$
\ddot{\mathbf{R}}_{app} + 2\Omega\times\dot{\mathbf{R}}_{app} + \Omega\times(\Omega\times\mathbf{R}_{app}) = -\frac{GM}{\left|\mathbf{R}_{app}\right|^3}\mathbf{R}_{app} + \mathbf{a}_{perturb}(\mathbf{R}_{app}, t) + \mathbf{a}_{res}(t) \tag{6}
$$

where $\mathbf{a}_{res}$(t) is a residual acceleration needed to satisfy the equality (since the Chebyshev series only approximates the solution.) The goal will be to find the series coefficients that make the residual acceleration as small as possible for a given maximum degree D.

NONLINEAR OPTIMIZATION OF THE RESIDUAL ACCELERATION

A practical approach to optimizing the residual acceleration begins by evaluating the residual acceleration over a set of $N > D$ constraint points $\{t_n \mid n=1,\ldots,N\}$ covering the interval $[0,T]$, yielding an over-determined system of $3N$ scalar equations with $3(D+1)$ scalar unknowns[1]. These equations cannot be solved in general, but they can be solved in a least-squares sense (minimizing the root-mean-squared residual acceleration over the set of constraint points). An iterative nonlinear optimization algorithm must be used because the residual acceleration is a nonlinear function of the Chebyshev coefficients even without the influence of the perturbations.

The set of Chebyshev coefficients are organized into a $3(D+1)$-dimensional vector $\mathbf{C}$ so that $\mathbf{C}_{3d+i}$ gives the i^{th} component (i = 1, 2 or 3) of $\mathbf{c}_d$. The estimate of the optimal vector of Chebyshev coefficients is refined each iteration. The algorithm is explained step-by-step below.

[1] Our product line orbit integrators use from 1.5 to 2 times the maximum degree for the number of constraint points.

LINEARIZING ABOUT THE ESTIMATE

At each constraint point t_n, the residual acceleration function is linearized about the current estimate for the optimal Chebyshev coefficients

$$\left(\mathbf{a}_{res}(t_n;\mathbf{C}+\Delta\mathbf{C})\right)_i \cong \left(A\cdot\Delta\mathbf{C}-\mathbf{B}\right)_{i+3n} \tag{7}$$

where $\mathbf{B}$ is a $3N$-dimensional vector

$$\mathbf{B}_{i+3n} = -\left(\mathbf{a}_{res}(t_n;\mathbf{C})\right)_i \tag{8}$$

and A is a $3N \times 3(D+1)$-dimensional matrix

$$\begin{aligned} A_{i+3n,j+3d} &= \frac{\partial\left(\mathbf{a}_{res}(t_n;\mathbf{C})\right)_i}{\partial\mathbf{C}_{j+3d}} \\ &= -W_{i,j}\cdot\frac{\partial\mathbf{R}}{\partial\mathbf{C}_{j+3d}} + \frac{\partial\left(\mathbf{a}_{res}\right)_i}{\partial\dot{\mathbf{R}}}\cdot\frac{\partial\dot{\mathbf{R}}}{\partial\mathbf{C}_{j+3d}} + \frac{\partial\left(\mathbf{a}_{res}\right)_i}{\partial\ddot{\mathbf{R}}}\cdot\frac{\partial\ddot{\mathbf{R}}}{\partial\mathbf{C}_{j+3d}} \end{aligned} \tag{9}$$

W is the gravity gradient matrix

$$W = -\frac{\partial\mathbf{a}_{res}}{\partial\mathbf{R}} = -\frac{GM}{\left|\mathbf{R}\right|^3}\left(\mathbf{1}_{3\times3} - 3\frac{\mathbf{R}\otimes\mathbf{R}}{\left|\mathbf{R}\right|^2}\right) + \frac{\partial\mathbf{a}_{perturb}}{\partial\mathbf{R}} \tag{10}$$

The partial derivatives on the right-hand side of (9) are computed from either the Chebyshev expansion formula (2) and its derivatives or the vector equation of motion (1). Only the perturbation term in equation (10) is difficult to calculate. Fortunately, it may be neglected as in most nonlinear optimization problems, moderate inaccuracy in the gradient matrix A can be tolerated. This approximation will not affect the calculation of the residual accelerations $\mathbf{B}$; at worst, it will slow the convergence of the algorithm.

CORRECTING THE ESTIMATE

The method of Singular Value Decomposition (SVD)[6] is used to find the correction $\Delta\mathbf{C}$ which minimizes the sum of the squares of the linearized residual acceleration evaluated at each constraint point: $\left\|A\cdot\Delta\mathbf{C}-\mathbf{B}\right\|^2$. The SVD method gives a decomposition of A of the form

$$\begin{aligned} A &= U\cdot Diag[\mathbf{w}]\cdot V^T \\ U^TU &= V^TV = \mathbf{1}_{3(D+1)\times3(D+1)} \end{aligned} \tag{11}$$

with U being $3N \times 3(D+1)$, V being $3(D+1) \times 3(D+1)$, and $\mathbf{w}$ containing the $3(D+1)$ singular values of A.

The matrix A will have a six-dimensional null space corresponding to the fact that six initial conditions need to be supplied to find a unique solution to the vector equation of motion (1). The null space manifests itself in the SVD method by the appearance of exactly six small elements among the singular values. The term "small" is taken to mean that the ratio $|\mathbf{w}_n|/\max|\{\mathbf{w}\}|$ approaches $10^{\text{-digits-of-precision}}$, meaning that $\mathbf{w}_n$ is indistinguishable from zero. The vector $\mathbf{w}^{-1}$ is formed by treating these six "small" singular values differently from the other "large" singular values:

$$\left(\mathbf{w}^{-1}\right)_n = \begin{cases} 1/\mathbf{w}_n & \text{for the "large" values of } \mathbf{w}_n \\ 0 & \text{for the "small" values of } \mathbf{w}_n \end{cases} \tag{12}$$

The SVD solution for $\Delta\mathbf{C}$ is then

$$\Delta\mathbf{C} = V \cdot Diag[\mathbf{w}^{-1}] \cdot U^T \cdot \mathbf{B} \tag{13}$$

The estimate for the optimal Chebyshev coefficients could simply be corrected by $\Delta\mathbf{C}$; however, the initial conditions need not be respected by the coefficient vector $\mathbf{C} + \Delta\mathbf{C}$. The SVD solution for $\Delta\mathbf{C}$ is not a unique solution, as any linear combination of vectors in the null space of A may be added to it. The null space of A is spanned by the columns of V corresponding to the six "small" singular values. These null space vectors are gathered to form a $3(D+1)$ 6-dimensional matrix P, so that the corrected coefficients are of the form $\mathbf{C} + \Delta\mathbf{C} + P \cdot \mathbf{q}$, for an arbitrary 6-dimensional vector $\mathbf{q}$. The next step is to solve for $\mathbf{q}$ so that the initial conditions $\mathbf{R}(0) = \mathbf{R}_0$ and $\dot{\mathbf{R}}(0) = \mathbf{V}_0 - \mathbf{\Omega} \times \mathbf{R}_0$ are respected. For this purpose, it is convenient to define a 6 $3(D+1)$-dimensional matrix holding the values for the Chebyshev polynomials and their derivatives at the initial time:

$$Z = \begin{pmatrix} T_0(-1)\mathbf{1}_{3\times3} & T_1(-1)\mathbf{1}_{3\times3} & T_2(-1)\mathbf{1}_{3\times3} & \ldots & T_D(-1)\mathbf{1}_{3\times3} \\ \frac{2}{T}T_0'(-1)\mathbf{1}_{3\times3} & \frac{2}{T}T_1'(-1)\mathbf{1}_{3\times3} & \frac{2}{T}T_2'(-1)\mathbf{1}_{3\times3} & \ldots & \frac{2}{T}T_D'(-1)\mathbf{1}_{3\times3} \end{pmatrix} \tag{14}$$

($T_n(-1) = (-1)^n, T_n'(-1) = -n^2(-1)^n$ follows from equations (5)) so that

$$Z \cdot \left(\mathbf{C} + \Delta\mathbf{C} + P \cdot \mathbf{q}\right) = \begin{pmatrix} \mathbf{R}(0) \\ \dot{\mathbf{R}}(0) \end{pmatrix} \tag{15}$$

The solution for $\mathbf{q}$ is then

$$\mathbf{q} = \left(Z \cdot P\right)^{-1}\left(\begin{pmatrix} \mathbf{R}(0) \\ \dot{\mathbf{R}}(0) \end{pmatrix} - Z \cdot \left(\mathbf{C} + \Delta\mathbf{C}\right)\right) \tag{16}$$

and the estimate is updated so that the initial conditions are respected

$$\mathbf{C} \leftarrow \mathbf{C} + \Delta\mathbf{C} + P \cdot \mathbf{q} \tag{17}$$

CONVERGENCE TOWARD MINIMUM RESIDUAL ACCELERATION

The root-mean-squared residual acceleration over the constraint points is

$$\sigma_{res} = \sqrt{\frac{\|\mathbf{B}\|^2}{N}} \tag{18}$$

The process of correcting the estimate continues until σ_{res} is inferior to the desired objective. It may be that this objective is unobtainable with the specified maximum degree D, in which case, the loop will not terminate.

INITIALIZING THE ESTIMATE

A good first guess to start with is

$$\mathbf{C} = \begin{pmatrix} \mathbf{R}(0) + \frac{1}{2}\dot{\mathbf{R}}(0)T \\ \frac{1}{2}\dot{\mathbf{R}}(0)T \\ 0 \\ \ldots \\ 0 \end{pmatrix} \tag{19}$$

VALIDATION AGAINST THE KEPLER SOLUTION

The Kepler solution applies exactly in the absence of perturbations and is also a nontrivial exercise of the algorithm just described. For these reasons, it serves as a useful validation case. The orbit integrators in our product line have also been cross-validated against other programs; however, it is arguable which algorithm is being tested. Figure 1 compares the Chebyshev polynomial and Kepler solutions as functions of time for a slightly eccentric and slightly inclined nearly geostationary orbit. Chebyshev polynomial solutions are generated one day at a time, with the final values used as the initial values for the next day. Figure 2 shows how the accuracy of a one-day solution varies with the maximum degree.

SOLUTION LEAF STRUCTURES

Each solution in terms of Chebyshev polynomials is applicable only over a defined time span T. If a longer span is desired, the order of the Chebyshev series will need to grow as well. Alternatively, the longer span can be covered by a sequence of solution leaves.

The k^{th} leaf is applicable over the span $[\theta_{k\text{-}1}, \theta_{k\text{-}1} + T]$ and consists of a single vector of Chebyshev polynomial coefficients $\mathbf{C}^k$ and a transition time to the next leaf θ_k in the interval $[\theta_{k\text{-}1}, \theta_{k\text{-}1} + T]$.

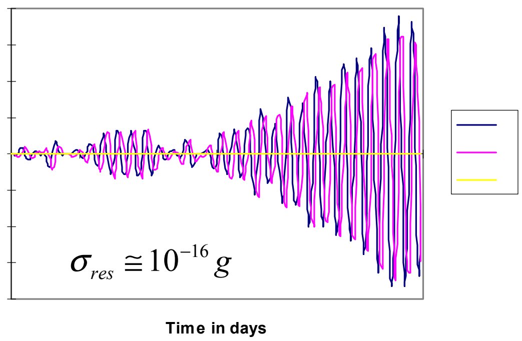

Figure 1. Comparison with Kepler Solution (Concatenated 1-day solutions, $D = 25$)

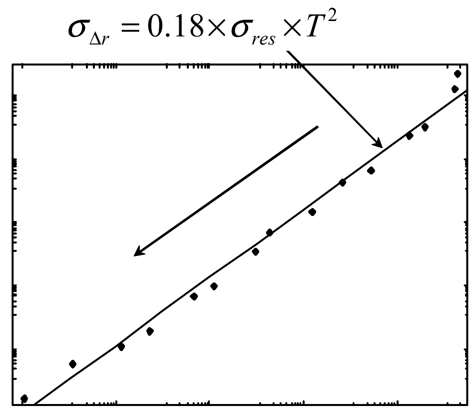

Figure 2. RMS Position Error vs. Residual Acceleration for a One-Day Solution

Solution leaf structures also help to solve a problem that is encountered when the solar radiation pressure is turned on or off upon leaving or entering eclipse. Because the illumination factor K in equation (2) changes discontinuously at eclipse entry and exit, the residual acceleration function is no longer a smooth function of the Chebyshev coefficients, making tight convergence difficult or impossible to achieve. Solution leaf structures solve this problem by representing the solar radiation pressure as always on or always off for the duration of the leaf. Leaves are created sequentially in time, and if an eclipse transition is encountered in the span of a leaf then the transition time to the next leaf is set equal to that eclipse transition time, as illustrated in Figure 3. Figure 4 is an illustration of the difference between a multi-leaf solution and its first leaf, showing the influence of the removal of solar radiation pressure in eclipse.

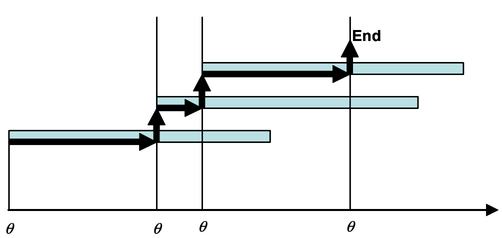

Figure 3. Leaf Structure Transitions for Eclipse

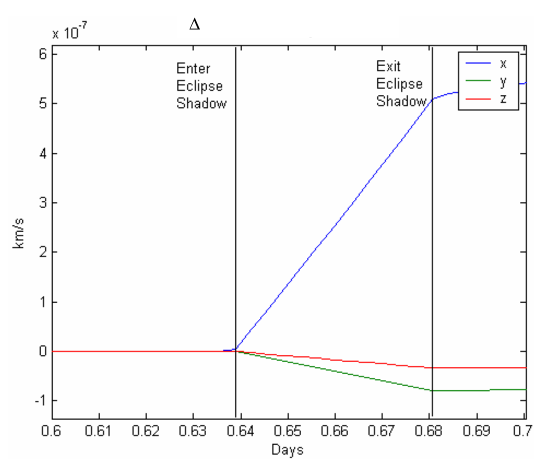

Figure 4. Influence of Eclipse on Leaves

SUMMARY

This paper has shown how the mathematics of optimization and estimation theory may be applied to solve differential equations of motion. Unlike traditional methods (*e.g.*, Runge-Kutta, Burlirsch-Stoer, or Predictor-Corrector methods), a true functional solution is found in terms of a Chebyshev polynomial series that may be easily evaluated out of chronological order. Arguably, the constraint points are analogous to the discrete time steps used by the traditional methods. Thinking in these terms, the Chebyshev polynomial series serves the purpose of providing an interpolation method between constraint points where the solution has been optimized.

The functional method developed in this paper is interesting in practice for several reasons. First it provides a solution with a numerical accuracy that may be tuned to the uncertainty of the forces acting on the body. This property should be useful for applications like formation flying. Second, it may be readily adapted to problems that are more challenging than the initial value problem. For example, to solve the split boundary value problem $\mathbf{R}(0) = \mathbf{R}_1$ and $\mathbf{R}(T) = \mathbf{R}_2$, one only needs to use these boundary values in place of the initial position and velocity respectively in equation (15), change the matrix Z defined in equation (14) to

$$Z = \begin{pmatrix} T_0(-1)\mathbf{1}_{3\times3} & T_1(-1)\mathbf{1}_{3\times3} & T_2(-1)\mathbf{1}_{3\times3} & \cdots & T_D(-1)\mathbf{1}_{3\times3} \\ T_0(+1)\mathbf{1}_{3\times3} & T_1(+1)\mathbf{1}_{3\times3} & T_2(+1)\mathbf{1}_{3\times3} & \cdots & T_D(+1)\mathbf{1}_{3\times3} \end{pmatrix} \tag{20}$$

and alter the initial guess given in equation (19) to

$$\mathbf{C} = \begin{pmatrix} \frac{1}{2}(\mathbf{R}_2 + \mathbf{R}_1) \\ \frac{1}{2}(\mathbf{R}_2 - \mathbf{R}_1) \\ 0 \\ \cdots \\ 0 \end{pmatrix} \tag{21}$$

REFERENCES

[1]Blancke, B, J. Carr, E. Lairy, F. Pomport and B. Pourcelot, "The Aerospatiale Meteosat Image Processing System (AMIPS)", First International Symposium on Scientific Imagery and Image Processing, L'Association Aéronautique et Astronautique de France, April 1995.

[2] Portable ESOC Package for Synchronous Orbit Control (PEPSOC) is a product of the European Space Operations Center (ESOC) in Darmstadt, Germany, http://www.esoc.esa.de/external/mso/oratos.html.

[3] Rosborough, G. W., “Gravitational Effects on Low-Earth Orbiters” in *Advances in the Astronautical Sciences*, Vol. 74, 1991.

[4] United States Naval Observatory, Naval Observatory Vector Astrometry Subroutines (NOVAS), http://aa.usno.navy.mil/software/novas/novas_info.html.

[5] Gradshteyn, I. S. and I. M. Rizhik, **Table of Integrals, Series, and Products**, Academic Press, N.Y., N.Y., 1980, pp. 1032-1033.

[6] Press, W. H., S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, **Numerical Recipes in C**, Second Ed., Cambridge University Press, Cambridge, U. K., 1992, pp 59-70.